UD-5011

Stepper Motor Driver System

OPERATION MANUAL

OBB™
Optical Building Blocks

PTI® Photon Technology International

MP MASTER
MA30229

UD-5011

Stepper Motor Driver System

OPERATION MANUAL

OBB™

Optical Building Blocks

PTI® Photon Technology International

Copyright © 1994 PTI. All Rights Reserved. OBB is a trademark and PTI is a registered trademark of Photon Technology International, Inc., 1 Deerpark Drive, Suite F, South Brunswick, NJ 08852 Phone: (908) 329-0910, Fax: (908) 329-9069. Specifications subject to change without notice.

PTI INSTRUMENT WARRANTY

Warranty Period and Extent

Photon Technology International (PTI) warrants that its instrument systems will meet stated specifications for a period of one (1) year. The warranty period will start on the date of shipment by PTI. In case of systems that include installation by PTI, this warranty will start from the date of installation or thirty (30) days after the shipping date, whichever is earlier. This warranty is in lieu of all other warranties, expressed or implied, including, without limitation, the implied warranties of merchantability and fitness for a particular purpose. PTI shall not be responsible for any liability, loss or damages, caused or alleged to be caused, by the system, as a result of use or operation including, without limitation, consequential damages and loss of profit.

Specific Exclusions and Limitations

1) It is recognized that the performance of consumable items will diminish as a function of use, and that it may be necessary to replace such items to restore the stated specifications. Consumable items (arc lamps, filters, cuvettes, lenses, etc.) are not covered by this warranty.

2) The original manufacturer's warranty will be maintained for major system components not manufactured by PTI (e.g. computers, printers, microscopes, cameras and components thereof).

3) Fiber optic bundles are not covered by this warranty.

4) The use of arc lamps not supplied by PTI (or approved in writing by PTI) will void PTI's warranty on all illuminator subsystem components.

5) If there is any evidence of physical contact with coated optics (e.g. fingerprints), the warranty on that item will be voided.

6) If the optical components are realigned by the customer without specific permission from PTI, the warranty will be voided. Please note that the customer is responsible for changing lamps and aligning the lamp after installation. The act of aligning the lamp will not void the warranty unless other exclusions are applicable (nos. 4 and 5).

7) Instrument systems that are not installed by PTI service personnel will not be warranted.

8) Moving systems to another site within a facility or to another location, without specific permission from PTI, will void this warranty.

9) Damage or loss caused by shipping is not covered by this warranty.

10) Damage caused by improper operation of the instrument will void this warranty.

11) Damage caused by equipment not purchased from PTI that is attached to the instrument is not covered by this warranty.

12) In case of systems that include installation as part of the original purchase, unpacking the instrument by anyone other than PTI personnel will void this warranty.

Warranty Returns

Please obtain a Return Authorization Number from our Service Department before shipping an item back to the PTI plant. Place the Return Authorization Number on the shipping label. Returned goods pertaining to the warranty will not be accepted without a Return Authorization Number. Returned goods should be shipped, freight prepaid, by the most appropriate manner.

Warranty Repairs

Warranty repairs will be done either at the customer's site or at the PTI plant, at our option. All service rendered by PTI will be performed in a professional manner by qualified personnel.

CONTENTS

1. SYSTEM OVERVIEW

Introduction 1
System Requirements 1
Motor Specifications 1
Board Layout 2
Connector Pinouts 3

2. INSTALLATION

Address Jumpers 5
Choosing a Base Address 6
Installing the Board 6

3. OPERATION

Introduction 7
Setup Mode 9
Control Mode 12

4. SOFTWARE CONTROL

Registers 15
Commands 19
Repeats 20
Pseudo Code 21
Sample Program 23
Control Register Memory Map 26

This page intentionally left blank.

1. SYSTEM OVERVIEW

INTRODUCTION

The UD5011 is a plug-in stepper motor controller/driver card for use in IBM PC/XT/AT computers and compatibles. The UD5011 will directly drive 2 stepper motors without external power supplies or level translators. The UD5011 comes with a diskette which contains a control program (PC5010.EXE) and an example BASIC program.

SYSTEM REQUIREMENTS

The UD5011 has been designed to work in IBM PC's and compatibles with ISA or EISA buses with bus clock speeds up to 12 MHz and zero wait states. The system must have a minimum of 256K of memory.

The UD5011 demands up to 2 amps of the +12 volt supply when both motors are rotating. Ensure that the computer's power supply can handle the extra demand.

MOTOR SPECIFICATIONS

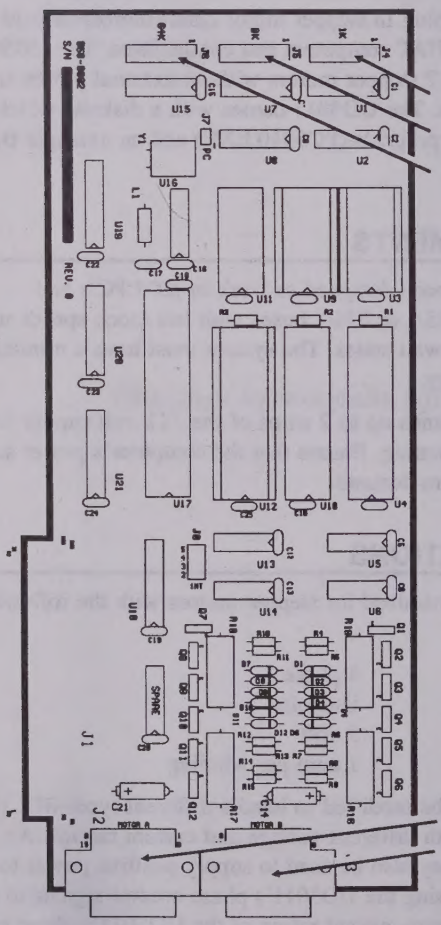
The UD5011 is optimized for stepper motors with the following specifications:

4 phase
Unipolar
5 volt
1 amp per winding

The UD5011 can be modified to handle different types of 4 phase stepper motors with different voltage and current ratings. An external supply may also be used to supply positive power to the motor windings using the UD5011's phase control signals to run the motor. The maximum current rating of the UD5011's phase control lines is 1 amp.

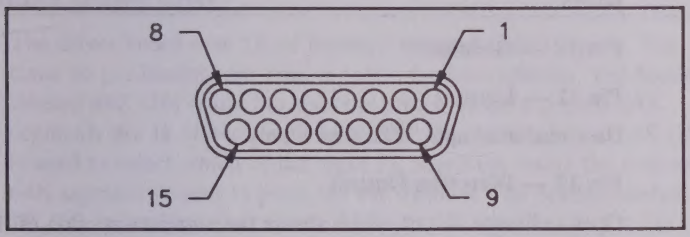
BOARD LAYOUT

The illustration below shows the component side of the 5011 board and points out the locations of the Memory Address Jumpers and the Motor Connectors.



CONNECTOR PINOUTS

The UD5010 card has two "D" style 15 pin female connectors, one for each motor. The illustration below shows the connector pin arrangement.



Pin 1 — Motor Power Output

5 volt 1 amp motor power supply.

Pin 2 — Upper Limit Input

An active low input; when pulled low tells the UD5011 that the upper limit has been reached and to stop the motor.

Pin 3 — Lower Limit Input

Active low input; when pulled low tells the UD5011 that the lower limit has been reached and to stop the motor.

Pin 4 — Callibration Input

Active low input; when pulled low tells the UD5011 that the calibration position has been reached and to stop the motor.

Pin 5 — Step Pulse Output

Active low open collector output; drops to ground potential for approximately .5 mSeconds for each motor step.

Pin 6 — Phase 1 Sink Input

Open collector phase 1 motor input capable of sinking 1.5 amps of current.

Pin 7 — Phase 3 Sink Input

Phase 3 motor input.

Pin 8 — Phase 5 Sink Input

Phase 5 motor input.

Pin 9 — +5 Volt Output

+5 volts output for powering external detection electronics, such as opto interrupters for detecting upper/lower limits or calibration points.

Pin 10 — Ground

Pin 11 — Input

User readable input. TTL levels only.

Pin 12 — Direction Output

Open collector output which shows the current direction of the motor.

Pin 13 — Output

User programmable open collector output.

Pin 14 — Phase 4 Sink Input

Phase 4 motor input.

Pin 15 — Phase 2 Sink Input

Phase 2 motor input.

2. INSTALLATION

ADDRESS JUMPERS

The driver board uses 1K of memory mapped address space. The three 16 pin headers are used to select the base address. The header labeled 64K (J6) is used to select in which of the top eight 64K segments the 1k window will be placed. The header labeled 8K (J5) is used to select which of the eight 8K segments inside the selected 64K segment is used to place the 1K window. The header labeled 1K (J4) is used to select one of the eight 1K segments inside the selected 8K segment.

Each header has eight jumper positions numbered 1 through 8. A jumper is installed in each header according to the values of three groups of three bits in the overall binary address as shown below:

ADDRESS LINES:

A19				A0
1 X X X	X X X X	X X X X	X X X X	X X X X
64K	8K	1K		

JUMPER POSITION	BIT VALUES
1	0 0 0
2	0 0 1
3	0 1 0
4	0 1 1
5	1 0 0
6	1 0 1
7	1 1 0
8	1 1 1

Example: If the base address of the board is to be D0000 (hex), first write out the values of the address lines:

A19				A0
1 1 0 1	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0

Using the Address Chart, the appropriate jumper positions are:

64k jumper	position 6
8k jumper	position 1
1k jumper	position 1

CHOOSING A BASE ADDRESS

The first 640K of system memory in an IBM PC/XT/AT computer is reserved for user RAM. The additional six 64K segments of the total 1 meg of memory that DOS can address are also reserved, but there are gaps that can be used by expansion cards. Be sure to check that none of these areas are used by other cards in your system. The following are some suggested locations for UD5011s:

UD5011 BOARD NO.	ADDRESS (HEX)	JUMPER POSITIONS (64K)	(8K)	(1K)
1	D0C00	6	1	4
2	D0800	6	1	3
3	D0400	6	1	2

INSTALLING THE UD5011

The UD5011 has a 31-pin edge connector and may be installed in any expansion slot. Be sure that all jumpers are properly set before installing the card on the motherboard.

> *WARNING:* Shut the power off to the computer and unplug it from the wall socket. Remove the cover and install the card as per instructions in your computer user manual. The card is static-sensitive and should be handled by the edges only.

3. CONTROL PROGRAM

3.1 INTRODUCTION

Supplied on the distribution diskette are both DOS- and Windows-based control programs. System requirements and installation instructions are indicated on the diskette label. The control programs can run up to three cards, controlling a total of 6 stepper motors.

The Windows-based program has complete context-sensitive Help. After installing MC and invoking the program, click on Help for all operation instructions.

The DOS-based program is significantly different from its Windows-based counterpart, and does not have a Help facility. The following instructions are provided for operating the UD-5011 on a PC that does not have the Windows operating system.

After installing the software, invoke the program by typing **PC5010** at the DOS prompt and pressing **Enter**. You will be presented with a screen that has a two-line by 40 character message display area and graphic representations of various keys. The message area is divided into three parts: Part 1 is the first 20 characters on the top line and shows the currently selected motor, the motor position, position units, and the direction of motion; Part 2 is the last 20 characters of the top line and shows key entry information; Part 3 is the bottom line and displays the current operation of function keys **F1** to **F5**. For modes that have more than 5 functions, **F5** will be labeled "more" and pressing **F5** will then display the other options available. These function options are looped, so that the initial function options can be recalled by repeatedly pressing the **F5** key. **F10** will terminate the program and return control to DOS.

To execute an operation, first press the appropriate function key (**F1-F5**), then make a numerical entry if required; finally, press the **Enter** key. The operation is not executed until **Enter** is pressed; entries made prior to pressing **Enter** can be cancelled by pressing **Delete** or by pressing another function key (**F1-F5**). The keys represented on the display relate to keyboard keys as follows:

Display Key	Keyboard Key	Function Invoked
F1-F5	Function Keys 1-5	Function Shown in Message Display
0-9	Number Keypad Keys	Numeric Entries
.	Decimal Point Key	Numeric & Key Separator
±	+ Key	Motor Direction Control
Del	Delete Key	Delete Entry
Mod	Letter "M"	Mode Selection
Ent	↵ Enter Key	Command Execution

Optionally, functions may be activated with a mouse by placing the mouse cursor over the display key graphic and pressing the mouse button.

The program source code is written in Microsoft C and has been included to show how to control a UD5011 from software. Refer to Section 4 for details on registers and commands.

3.2 SETUP MODE

The PC5010 program has two operating modes for each motor. Mode 1 is the SETUP mode and Mode 2 is the CONTROL mode. Pressing **MOD** ("**M**" character key) will toggle between the two modes. When invoked, the program comes up in the SETUP mode. All motor control parameters are initiated in the SETUP mode. Note that MOTOR, LOCATN and STEP V must be set before any other commands can be issued.

MOTOR

A MOTOR command simultaneously selects and labels which motor to set up and control. For each UD5011 board, the motors are identified as a logical A or B. The display, however, labels the motors with user-selected numbers.

When the program is invoked, the default motor numbers are 1 and 2 for A and B respectively, and the default control selection is motor 1. If only a single UD5011 is present, a MOTOR command is necessary only to select motor 2 first; otherwise, proceed to the LOCATN command. If there is more than one UD5011 board, a MOTOR command is necessary to re-label and select motors controlled by the other boards. To change the motor label/ selection, press the MOTOR function key and enter a single-digit integer. An odd number will label/select motor A, and an even number will label/select motor B. The motors are automatically labeled in pairs, i.e. if 3 is entered, motor A will be labeled 3 and motor B will be 4; if a 6 is entered, motor A will be 5 and B will be 6. These labels will apply to the motors controlled by the board identified with the LOCATN command that follows, and will remain in effect until the program is terminiated. After the initial labeling funtion, the MOTOR command is used only for motor selection.

> ***NOTE:*** Only one motor on one board can be controlled at a time.

LOCATN

LOCATN tells the program the memory location of the UD5011 by entering the 1K, 8K and 64K jumper positions respectively.

For example, If the 1K jumper position is 7, the 8K jumper position is 7 and the 64K jumper position is 6, type 776 followed by the Enter key. When more than one UD5011 is present, LOCATN is used to select the board to which commands will be issued, thereby determining the motor pairs controlled; see the MOTOR function for details on motor labeling and selection. A MOTOR command must be issued before the LOCATN can be entered if the default motor settings are incorrect.

> *NOTE:* LOCATN *must* be entered before any other commands can be issued.

STEP V

STEP V is the number of user units per step (to three decimal places). As an example, when used to drive a scanning monochromator, each motor step might correspond to 0.625 nanometers (user units), in which case STEP V = 0.625. Once STEP V has been specified, all subsequent position and dynamic parameter inputs as well as display message outputs will be in user units. The default value is 1, which sets the user units equal to motor steps.

> *NOTE:* The STEP V value must be greater than 0.

STEP M

STEP M sets the stepping mode. Enter .5 for half stepping or 1 for full stepping.

POSITN

POSITN sets the PC5010 position display equal the actual position of whatever the motor is driving. The POSITN value is entered in user units; refer to the STEP V function for details.

UNITS

UNITS allows the input of an ASCII string up to 4 characters to be used in the display to label the motor position in user units. For example, "NM" may be entered to define the position as being in nanometers.

EFSS

If the stepper motor is connected to a high inertia load, ramping may be required to attain the selected speed. The EFSS value sets the maximum Error Free Start Stop speed of the device that the stepper motor is driving. If the selected speed is faster than the EFSS, the motor will be started at the EFSS speed and ramped up to the selected speed. The same process is applied when stopping. The EFSS value is entered in user units per second; refer to the STEP V function for details.

RAMP L

RAMP L sets the length in user units of the acceleration/ deceleration ramp from starting (EFSS) speed to full speed and from full down to stopping (EFSS) speed. The RAMP L value is entered in user units; refer to the STEP V function for details.

SLEW S

SLEW S sets the speed at which to slew the motor when the SLEW function in the CONTROL mode is used. This speed is also used by the MOVE C function. The SLEWS value is entered in user units per second; refer to the STEP V function for details.

UPPERL

UPPERL sets the upper limit that the motor is allowed to move to before stopping. The UPPERL value is entered in user units; refer to the STEP V function for details.

LOWERL

LOWERL sets the lower limit that the motor is allowed to move to before stopping. The LOWERL value is entered in user units; refer to the STEP V function for details.

MOVE C

MOVE C will move the motor in the currently selected direction and at the speed entered with the SLEW S function until the calibration input line of the motor is brought to ground potential. See Section 1.5 for identification of this line. The STOP function can be used to interrupt the move.

3.3 CONTROL MODE

All motion controls are executed in the CONTROL mode. All functions involving movement or position are entered in user units as detailed in Section 3.2, STEP V.

MOVE A

MOVE A sets a numerical position to which the motor will move. Speed of travel is set by the SPEED function. The MOVE A value is entered in user units; see Section 3.2, STEP V for details.

MOVE R

MOVE R sets a numerical distance for the motor to travel. Direction is toggled using ± (the + key) and current direction is shown on the display. Speed is set by the SPEED function. The MOVE R value is entered in user units; see Section 3.2, STEP V for details.

SPEED

SPEED sets the rate of motion used in the MOVE A and MOVE R functions. Speed is entered in user units per second; see Section 3.2, STEP V for details.

STOP

STOP haults rotation of the motor.

SLEW

SLEW alternately starts and stops the slewing operation. The motor moves at the rate set with the SLEW S function in the SETUP mode. Direction of slewing is toggled using ± (the + key) and current direction is shown on the display. Do *not* use the STOP function to stop a slewing motor.

JOG

This function will jog or move the motor a single *motor step* in the current direction. Note that this function operates independently of the STEP V factor; that is, when the motor is jogged one step, the position display will change according to the STEP V value entered.

For example, if STEP V=0.625 and the current location is 0.625, jogging the motor forward one step will result in a move to location 1.250. The direction of jogging is toggled using ± (the + key) and current direction is shown on the display.

REPEATS

This function sets the number of times a MOVE A or a MOVE R function will be repeated. For more details on repeats, refer to Section 4.3.

DELAY

This function sets the delay in seconds between MOVE A or MOVE R repeats. For more details on repeats, refer to Section 4.3.

BACKL

BACKL sets the backlash correction distance when a MOVE A function is repeated. For more details on repeats, refer to Section 4.3. The BACKL value is entered in user units; see Section 3.2, STEP V for details.

RETSPD

RETSPD sets the return to starting point speed when a MOVE A function is repeated. For more details on repeats, refer to Section 4.3. The RETSPD value is entered in user units per second; see Section 3.2, STEP V for details.

This page intentionally left blank.

4. SOFTWARE CONTROL

The UD5011 can be controlled directly from user-written software. A sample program written in BASIC is listed in Section 4.5. Note that when controlling the UD5011 in this manner, motor movement and position is based on motor steps, not user units.

4.1 REGISTERS

There are a total of 40 registers, 19 for each motor (A and B) and 2 for commands. These registers vary in width from 1 byte to 4 bytes. The location with respect to the base address and size in bytes for each register is listed in Section 4.6. Multiple byte-wide registers are formatted such that the LSB is the first byte. For registers that are 4 bytes wide, only the 3 least significant bytes are used; the fourth byte is ignored.

SPEED Register

This register sets the stepping speed of the motor. The units are in 1/10000 seconds per step. For example, placing the number 1000 in this register would set the time between steps to .1 seconds or 10 steps per second.

DIRECTION Register

This register sets the direction of rotation for the motor. Placing 00 in this register sets clockwise rotation and 01 sets counterclockwise rotation.

UPPERLIMIT Register

This register sets the upper limit of travel in motor steps.

LOWERLIMIT Register

This register sets the lower limit of travel in motor steps.

MOVETO Register

This register is used in conjunction with the MOVE ABSOLUTE command. If a MOVE ABSOLUTE command isissued, the motor will rotate until the MOVETO position is reached.

DISTANCE Register

This register is used in conjunction with the MOVE RELATIVE command. If a MOVE RELATIVE command is issued, the motor will rotate the number of steps specified in this register in the direction specified in the DIRECTION register.

POSITION Register

This register contains the current position of the motor. It should be initialized to the position of whatever the motor is driving.

MOVING Register

This flag shows if a motor is currently rotating or stationary. A value of 00 present indicates the motor is stationary and a value of 01 indicates the motor is rotating.

EFSS Register

This register should be initialized to the maximum Error Free Start Stop stepping speed. This is the rate that a motor driving a given load can start of stop without missing steps. Units are the same as the SPEED register.

STPFSS Register

If the selected SPEED is greater than the EFSS, the motor speed will be ramped. This register is used to specify the number of steps to go from starting EFSS speed to full selected speed. Setting this register to 00 will tell the driver not to ramp the speed.

NUMBEROFPHASES Register

This register sets the number of phases for the motor it is driving. Currently only a value of 4 is valid.

STEPMODE Register

This register is used to specify FULL stepping or HALF stepping mode. 00 enables full stepping and 01 enables half stepping.

OUTPUT Register

This register is set to either 00 or 01 with the SET OUTPUT command.

INPUT Register

When the READ INPUT command is issued, the Input pin level will set this register to either 00 or 01.

REPEATS Register

This register is used in conjuction with the MA or MR commands. When this register reads 000, the MA and MR commands work normally; when numbers from 002 to 256 are present, the MA and MR commands will repeat the number of times specified. For more details on repeats, refer to Section 4.3.

DELAY Register

The number in this register sets the delay or pause between repeats of an MA or MR command. The units are the sames as those for SPEED. For more details on repeats, refer to Section 4.3.

BACKLASH Register

This register is used when repeating an MA command multiple times (REPEAT register is greater than 2). For more details on repeats, refer to Section 4.3.

RETURNSPEED Register

This register is the speed used to return to the starting position when repeating an MA command. The units are the sames as those for SPEED. For more details on repeats, refer to Section 4.3.

ERROR Register

This register will contain an error code when the following conditions occur:

00 — No error
01 — Upper limit has been reached
02 — Lower limit has been reached
03 — Calibration point has been reached
04 — Illegal/unknown command

COMMANDBUFFER Register

This register is where the command to be executed is placed. The command will not be executed until the COMMAND-EXEC register is written to.

COMMANDEXEC Register

This register will cause the command placed in the COMMANDBUFFER register to be executed:

"A" — A command for motor A only
"B" — A command for motor B only
"Z" — A command for motors A and B

After placing one of these characters in the COMMANDEXEC register, the register will read 00 when the command has been accepted.

4.2 COMMANDS

The following is a list of the commands that can be placed in the COMMANDBUFFER register:

ST — STOP motor

The motor(s) specified by the character put in the COMMAND-EXEC register is stopped. If motor is already stopped, this command is ignored.

JG — JOG motor

The motor(s) specified by the character put in the COMMAND-EXEC register is moved 1 step in the direction specified by the content of the DIRECTION register. The size of the step (FULL of HALF) is set by the content of the STEPMODE register.

MA — MOVE ABSOLUTE

The motor(s) specified by the character put in the COMMAND-EXEC register is moved to the position specified in the MOVETO register. The speed is set by the content of the SPEED register and the step size is set by the content of the STEPMODE register.

MR — MOVE RELATIVE

The motor(s) specified by the character put in the COMMAND-EXEC register is moved the number of steps specified in the DISTANCE register. The speed is set by the content of the SPEED register. The direction of motion is set by the content of the DIRECTION register, and the step size is set by the content of the STEPMODE register.

MC — MOVE TO CALIBRATION POINT

The motor(s) specified by the character put in the COMMAND-EXEC register is moved in the direction specified by the content of the DIRECTION register at the speed specified by the content of the SPEED register, and the step size is set by the content of the STEPMODE register. The motor will continue moving until the calibration input pin is brought down to ground potential.

OP — SET OUTPUT

The motor(s) specified by the character in the COMMAND-EXEC register sets the output pin to the logic level specified in the OUTPUT register.

IP — READ INPUT

The motor(s) specified by the character in the COMMAND-EXEC register has its INPUT register updated according to the state of the input pin.

4.3 REPEATS

When the MA command is executed, the motor will move from the current or starting position to the position specified in the MOVETO register at the speed set by the SPEED register. It will then pause for the length of time specified by the content of the DELAY register. When the delay time has elapsed, the motor will move back to the starting position. The return speed is set by the content of the RETURNSPEED register. If the BACKLASH register contains a value other than 0, the motor will be moved past the starting position by this value. At this point the original MA command is repeated.

In contrast, when the MR command is executed, the motor will move from the current position a distance set by the content of the DISTANCE register and in the direction set by the DIRECTION register. At the end of this move, the motor will pause for the length of time set by the content of the DELAY register. When this time has elapsed, the motor will move the distance specified by the DISTANCE register and in the direction set by the content of the DIRECTION register. This cycle will repeat the number of times specified in the REPEATS register.

4.4 PSEUDO CODE

The following example is a plain English program to demonstrate the sequence of steps required to control the UD5011. A prefix of A_ or B_ denotes motor A and motor B registers respectively and no prefix means that this register is common to both motors.

In the following section, the position is set to 1000, the step mode is set to HALF step, the direction is set for counterclockwise and the speed is set to 100 steps/second:

```
1   WRITE 1000 to A_POSITION register        ;Set position
2   WRITE 1 to A_STEPMODE register           ;Set half step
3   WRITE 1 to A_DIRECTION register          ;ccw direction
4   WRITE 500 to A_DISTANCE register         ;500 steps
5   WRITE 100 to A_SPEED register            ;100 steps/sec
6   WRITE MR to COMMANDBUFFER register ;Set command
7   WRITE A to COMMANDEXEC register          ;Move motor A
8   READ COMMANDEXEC register                ;Check if accepted
9   IF COMMANDEXEC = A GOTO 8                ;If not, wait
10  READ A_ERROR register                    ;Check if all OK
11  IF A_ERROR = 0 GOTO 13                   ;If no error, continue
12  EXIT                                     ;If error, exit
13  READ A_MOVING register                   ;Get motion flag
14  IF A_MOVING = 1 GOTO 13                  ;Wait until stopped
15  READ A_POSITION register                 ;Should read 500
```

In the following section the EFSS is set to 100 steps/sec and the ramp length is set to 50 steps. The motor is then programmed to move at 1000 steps/sec to a position of 1500:

```
16  WRITE 100 to A_EFSS register             ;Set to 100 steps/sec
17  WRITE 50 to A_STPFSS register            ;Set ramp to 50 steps
18  WRITE 10 to A_SPEED register             ;Set to 1000 steps/sec
19  WRITE 15000 to A_MOVETO register         ;
20  WRITE MA to COMMANDBUFFER register;Set command
21  WRITE A to COMMANDEXEC                   ;Execute command
22  READ COMMANDEXEC                         ;Wait for command
23  IF COMMANDEXEC = A GOTO 22               ;acceptance
```

While motor A is rotating, motor B will be initialized in the following section and programmed to move to a calibration point. The absolute value of the calibration point is 00:

```
24  WRITE 200 to B_SPEED register          ;50 steps/sec
25  WRITE 1 to B_DIRECTION register        ;ccw
26  WRITE MC to COMMANDBUFFER register ;Set command
27  WRITE B to COMMANDEXEC                 ;Execute command
28  READ COMMANDEXEC                       ;Wait for command
29  IF COMMANDEXEC = B GOTO 28             ;acceptance
30  READ B_MOVING                          ;Wait until calib. point
31  IF B_MOVING = 1 GOTO 30                ;has been reached
32  WRITE 0 to B_POSITION                  ;Set position to 00
```

4.5 SAMPLE BASIC PROGRAM

The following is a listing of the BASIC program BASDEMO.BAS on the disk supplied with the UD5011. It is an example of how to control the UD5010 with a BASIC program.

```
10 '************************************************************
20 ' In this example, motors A and B of a UD5011 whose jumpers are set
30 ' to 1 1 6 (Address D000hex), are moved back and forth continually.
40 ' Refer to your BASIC manual for explanations of the BASIC language
50 ' statements and refer to the UD5010 manual for UD5011 programming.
60 '
70 '************************************************************
80 '
90 CLS 2            'Clear screen
100 LOCATE 2,1
110 PRINT "Motor A position ="
120 PRINT "Motor B position ="
130 '
140 ' Define segment that the UD5011 is located in.
150 '
160 DEF SEG = &HD000
170 '
180 ' Make sure both motors are not moving
190 '
200 POKE 1020,ASC("S")     'Place "ST" command in command buffer
210 POKE 1021,ASC("T")
220 POKE 1023,ASC("Z")     'Execute command for both motors
230 '
240 ' Set position of motor A to 10000 and motor B to 20000
250 '   10000 = 2710hex
260 '   20000 = 4E20hex
270 '
280 POKE 789,&H10          'Set Position of motor A
290 POKE 790,&H27
300 POKE 791,&H0
310 POKE 840,&H20          'Set Position of motor B
320 POKE 841,&H4E
330 POKE 842,&H0
340 '
350 ' Load SPEED register of both motors with 20 for a speed of:
360 '    .0001 x 20 = .0020 sec/step = 500 steps per second
370 '
380 POKE 768,20            'Set Speed of motor A
390 POKE 769,0
400 POKE 770,0
410 POKE 819,20            'Set Speed of motor B
420 POKE 820,0
430 POKE 821,0
440 '
```

```
450 ' Load DISTANCE registers of both motors with 1000, which is
460 ' equal to 03E8 hex.
470 '
480 POKE 785,&HE8        'Set Distance of travel for motor A
490 POKE 786,&H3
500 POKE 787,0
510 POKE 836,&HE8        'Set Distance of travel for motor B
520 POKE 837,&H3
530 POKE 838,0
540 '
550 ' Load DIRECTION register of motor A w/ cw direction and DIRECTION
560 ' register of motor B with ccw direction.
570 '
580 POKE 772,0          'Set Direction of motor A to cw
590 POKE 823,1          'Set Direction of motor B to ccw
600 '
610 ' Set EFSS registers of both motors to 100. This will set starting
620 ' speed of motors to 100 x .0001 = .01 sec/step = 100 steps/sec
630 '
640 POKE 794,100        'Set EFSS of motor A
650 POKE 795,0
660 POKE 796,0
670 POKE 845,100        'Set EFSS of motor B
680 POKE 846,0
690 POKE 847,0
700 '
710 ' Set STPFSS registers of both motors to 50. This will set ramp
720 ' length to 50 steps.
730 '
740 POKE 798,50         'Set Ramp length of motor A
750 POKE 849,50         'Set Ramp length of motor B
760 '
770 ' Set the stepping mode of both motors to HALF STEP.
780 '
790 POKE 801,1          'Set Stepping mode of motor A
800 POKE 852,1          'Set Stepping mode of motor B
810 '
820 ' Place the MR (Move Relative) command in the command buffer.
830 '
840 POKE 1020,ASC("M")
850 POKE 1021,ASC("R")
860 '
870 ' Execute the MR command for both motors
880 '
890 POKE 1023,ASC("Z")
900 '
910 ' Call routine to display position
920 '
930 GOSUB 1120
940 '
950 ' Wait for 3 second before moving again
960 '
970 START = TIMER
980 IF TIMER (START+3) GOTO 980
990 '
```

```
1000 ' Reverse direction of motors and move again
1010 '
1020 A = PEEK(772)       'Get current direction of motor A
1030 B = PEEK(823)       'Get current direction of motor B
1040 POKE 772,B          'Set direction of motor A
1050 POKE 823,A          'Set direction of motor B
1060 GOTO 890
1070 '
1080
'=========================================
1090 ' This subroutine displays the motors position for as long as
1100 ' as they are moving. When motors stop it returns to caller.
1110 '
1120 IF PEEK(1023) = ASC("Z") GOTO 1120 'Wait til command is accepted
1130 FINISH = 0          'Set finish flag to NOT FINISHED
1140 IF (PEEK(793)+PEEK(844)) 0 GOTO 1160 'Check if motors are stopped
1150 FINISH = 1          'If yes, set finished flag to FINISHED
1160 ALOCATION = PEEK(789)   'Calculate motor A's position
1170 ALOCATION = ALOCATION*256 + PEEK(790)
1180 ALOCATION = ALOCATION*256 + PEEK(791)
1190 '
1200 BLOCATION = PEEK(840)   'Calculate motor B's position
1210 BLOCATION = BLOCATION*256 + PEEK(841)
1220 BLOCATION = BLOCATION*256 + PEEK(842)
1230 '
1240 LOCATE 2,20         'Show motor A's position
1250 PRINT ALOCATION
1260 LOCATE 3,20         'Show motor B's position
1270 PRINT BLOCATION
1280 '
1290 IF FINISH = 0 GOTO 1140 'If finish flag NOT FINISHED, loop again
1300 RETURN              'If finish flag is FINISHED, done
```

4.6 CONTROL REGISTER MEMORY MAP

The location, name and size for each register is shown in the following table. A prefix of A_ or B_ in front of the register name denotes motor A and motor B registers respectively and no prefix means that this register is common to both motors.

> *NOTE:* Locations with base address offsets 0 to 2FE and 366 to 3FB are reserved and should *not* be used.

(hex)	Offset from Base Address (decimal)	Width Name(bytes)
0-2FE	0-767	***reserved – do not use***767
300	768	A_SPEED4
304	772	A_DIRECTION1
305	773	A_UPPERLIMIT4
309	777	A_LOWERLIMIT4
30D	781	A_MOVETO4
311	785	A_DISTANCE4
315	789	A_POSITION4
319	793	A_MOVING1
31A	794	A_EFSS4
31E	798	A_STPFSS2
320	800	A_NUMBEROFPHASES1
321	801	A_STEPSIZE1
322	802	A_OUTPUT1
323	803	A_INPUT1
324	804	A_REPEATS2
326	806	A_DELAY4
32A	810	A_BACKL4
32E	814	A_RETURNSPEED4
332	818	A_ERROR1

(hex)	Offset from Base Address (decimal)	Width Name(bytes
333	819	B_SPEED4
337	823	B_DIRECTION1
338	824	B_UPPERLIMIT4
33C	828	B_LOWERLIMIT4
340	832	B_MOVETO4
344	836	B_DISTANCE4
348	840	B_POSITION4
34C	844	B_MOVING1
34D	845	B_EFSS4
351	849	B_STPFSS2
353	851	B_NUMBEROFPHASES1
354	852	B_STEPSIZE1
355	853	B_OUTPUT1
356	854	B_INPUT1
357	855	B_REPEATS2
359	857	B_DELAY4
35D	861	B_BACKL4
361	865	B_RETURNSPEED4
365	869	B_ERROR1
366-3FB	870-1019	***reserved – do not use***149
3FC	1020	COMMANDBUFFER2
3FF	1023	COMMANDEXEC1

This page intentionally left blank.

OBB™

Optical Building Blocks

PTI® Photon Technology International

MP MASTER

MA30229